**OPEN ENDED PROJECT**

on

**DIGITAL IMAGE PROCESSING**

**Submitted to:**

Amity School of Engineering and technology,

Noida

**Submitted to: Submitted by:**

Ms. Anupama Bahn ANURAG TRIPATHI

A12405117001,

7 ECE 2 -X ,B.TECH, ASET



**AMITY UNIVERSITY UTTAR PRADESH**

**GAUTAM BUDDHA NAGAR**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Content** | **Page No.** |
| 1 | AIM AND OBJECTIVE | 3 |
| 2 | APPARATUS AND SOFTWARE USED | 3 |
| 3 | THEORY | 3 |
| 4 | CODE WITH RESULT | 4 |
| 5 | CONCLUSION | 14 |

1. **AIM AND OBJECTIVE:**
2. To perform the unsharp masking by convolving the input image f(x, y)

the average filter and save your output image and then perform the difference between original and output image. Create another enhanced image from your output image such that the enchanted image is the summation of original input image and the difference image. And repeat the same procedure using the Laplacian filter and compare the result of both.

1. Take any grey-scale image and apply Laplacian operator on it and then apply the Gaussian filter and then do the vice versa and also check which among these two gives better results.
2. **APPARTUS AND SOFTWARE USED:**

MATLAB 2020a

1. **THEORY:**

The image we consider here is a standard grey level image, ‘The Cameraman’. The image is shown below:



Figure1: The Cameraman Image

**Average** (or mean) filtering is a method of 'smoothing' images by reducing the amount of intensity variation between neighbouring pixels. The average filter works by moving through the image pixel by pixel, replacing each value with the average value.

A **Laplacian filter** is an edge detector used to compute the second derivatives of an image, measuring the rate at which the first derivatives change. This determines if a change in adjacent pixel values is from an edge or continuous progression.

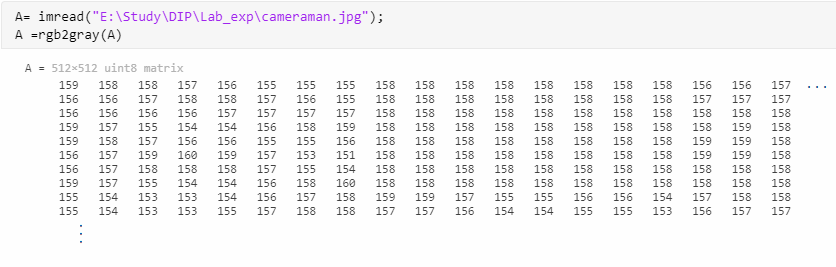
**Grayscale** is a range of monochromatic shades from black to white. Therefore, a grayscale image contains only shades of grey and no colour. Many image editing programs allow you to convert a colour image to black and white, or grayscale. This process removes all colour information, leaving only the luminance of each pixel.

A **Gaussian filter** is a linear filter. It's usually used to blur the image or to reduce noise. The Gaussian filter alone will blur edges and reduce contrast. The Median filter is a non-linear filter that is most commonly used as a simple way to reduce noise in an image.

1. **CODE WITH RESULT:**
2. To perform the unsharp masking by convolving the input image f(x,y) the average filter and save your output image and then perform the difference be original op image. Create another enhanced image from your op image such that the enchanted image is the summation of original input image and the difference image. And repeat the same procedure using the Laplacian filter and compare the result of both.

A= imread("E:\Study\DIP\Lab\_exp\cameraman.jpg");

A =rgb2gray(A)



figure;

b = double(A);

subplot(2,3,1)

imshow(A);

title('original image');



h2=1/25\*ones(5,5);

b1=conv2(double(A),double(h2),'same');

figure;

imshow(uint8(b1));

title('average filter(5\*5)');



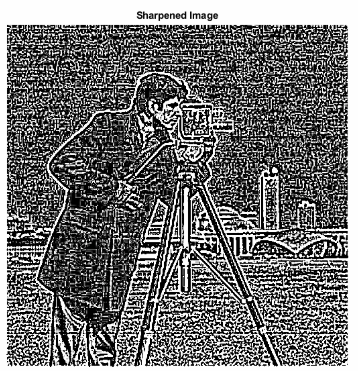
% Unsharp masking step

sh = b - b1;

figure;

imshow(sh);

title('Sharpened Image');



% Getting back our original image

org = sh + b1;

figure;

imshow(uint8(org));

title("Original = Sharpened + L.P.F");



1. Take any grey-scale image and apply Laplacian operator on it and then apply the Gaussian filter and then do the vice versa and also check which among these two gives better results.

% Laplacian Filter

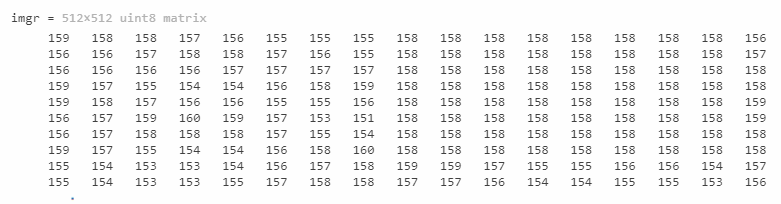
close all;

clear all;

clc;

imgr= imread("E:\Study\DIP\Lab\_exp\cameraman.jpg");

imgr =rgb2gray(imgr)



imshow(imgr);

title('Orignal Image')



figure

imd=im2double(imgr);

[r,c]=size(imd);

padded=zeros(r+2,c+2);

padded(2:513,2:513)=imd;

figure

mask=[1 1 1; 1 -8 1; 1 1 1];

for i=2:r

for j=2:c

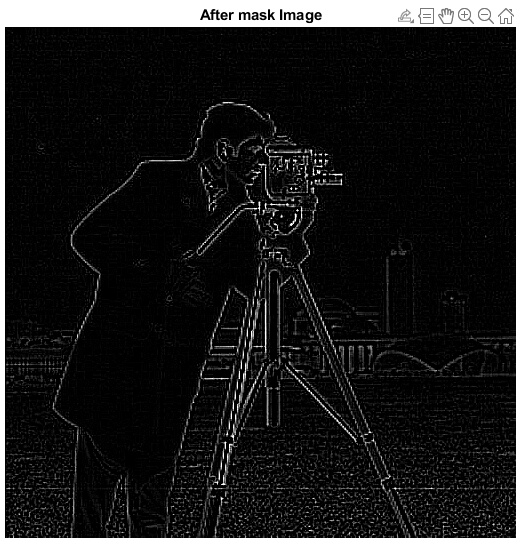
out(i-1,j-1)=mask(1,1)\*padded(i-1,j-1)+mask(1,2)\*padded(i-1,j)+mask(1,3)\*padded(i-1,j+1)+mask(2,1)\*padded(i,j-1)+mask(2,2)\*padded(i,j)+mask(2,3)\*padded(i,j+1)+mask(3,1)\*padded(i+1,j-1)+mask(3,2)\*padded(i+1,j)+mask(3,3)\*padded(i+1,j+1);

end

end

imshow(out);

title('After mask Image')



% Gaussian Filter

im2=imread("E:\Study\DIP\Lab\_exp\cameraman.jpg");

imshow(im2);

title('Orignal Image')



figure

imgr2=out;

imshow(imgr);

title('Gray Image')

figure

imd2=im2double(imgr2);

noi2=imnoise(imd2,'gaussian');

[r,c]=size(noi2);

padded=zeros(r+2,c+2);

padded(2:512,2:512)=noi2;

figure

mask=[1 2 1; 2 4 2; 1 2 1];

for i=2:r

for j=2:c

out2(i-1,j-1)=mask(1,1)\*padded(i-1,j-1)+mask(1,2)\*padded(i-1,j)+mask(1,3)\*padded(i-1,j+1)+mask(2,1)\*padded(i,j-1)+mask(2,2)\*padded(i,j)+mask(2,3)\*padded(i,j+1)+mask(3,1)\*padded(i+1,j-1)+mask(3,2)\*padded(i+1,j)+mask(3,3)\*padded(i+1,j+1);

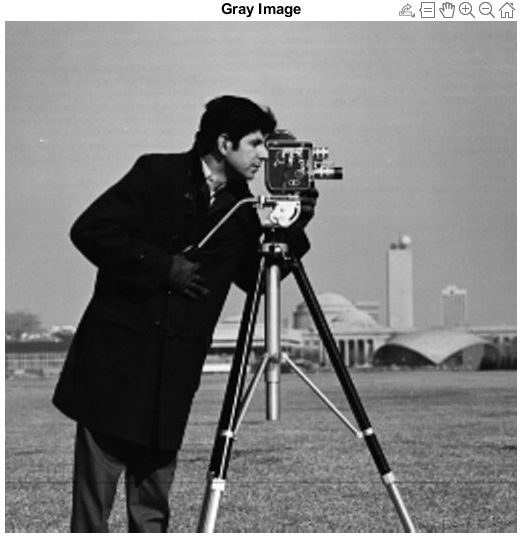
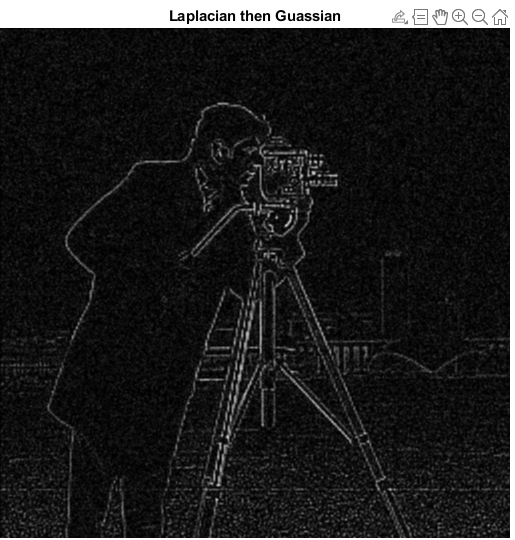
end

end

out2=out2/16;

imshow(out2);

title('Laplacian then Guassian')

% Gaussian Filter

close all;

clear all;

clc;

im=imread("E:\Study\DIP\Lab\_exp\cameraman.jpg");

im =rgb2gray(im)

imshow(im);

title('Orignal Image')

figure

imgr=im;

imshow(imgr);

title('Gray Image')

figure

imd=im2double(imgr);

noi=imnoise(imd,'gaussian');

figure

[r,c]=size(noi);

padded=zeros(r+2,c+2);

padded(2:513,2:513)=noi;

imshow(padded);

title('Padded Image')

figure

mask=[1 2 1; 2 4 2; 1 2 1];

for i=2:r

for j=2:c

out(i-1,j-1)=mask(1,1)\*padded(i-1,j-1)+mask(1,2)\*padded(i-1,j)+mask(1,3)\*padded(i-1,j+1)+mask(2,1)\*padded(i,j-1)+mask(2,2)\*padded(i,j)+mask(2,3)\*padded(i,j+1)+mask(3,1)\*padded(i+1,j-1)+mask(3,2)\*padded(i+1,j)+mask(3,3)\*padded(i+1,j+1);

end

end

out=out/16;

imshow(out);

title('Image after appling mask')

% Laplacian Filter

im=imread("E:\Study\DIP\Lab\_exp\cameraman.jpg");

figure

imgr=out;

imd=im2double(imgr);

[r,c]=size(imd);

padded=zeros(r+2,c+2);

padded(2:512,2:512)=imd;

figure

mask=[1 1 1; 1 -8 1; 1 1 1];

for i=2:r

for j=2:c

out(i-1,j-1)=mask(1,1)\*padded(i-1,j-1)+mask(1,2)\*padded(i-1,j)+mask(1,3)\*padded(i-1,j+1)+mask(2,1)\*padded(i,j-1)+mask(2,2)\*padded(i,j)+mask(2,3)\*padded(i,j+1)+mask(3,1)\*padded(i+1,j-1)+mask(3,2)\*padded(i+1,j)+mask(3,3)\*padded(i+1,j+1);

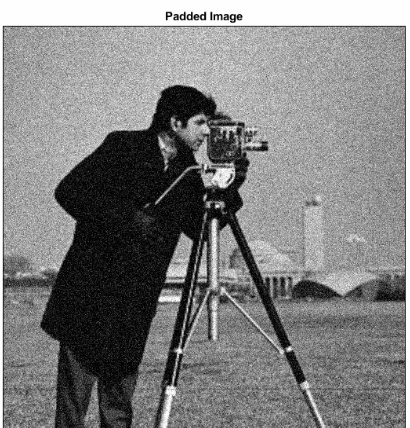
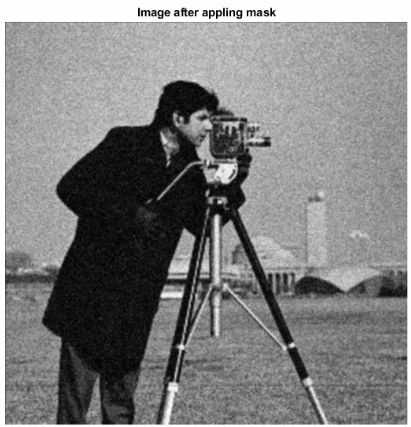
end

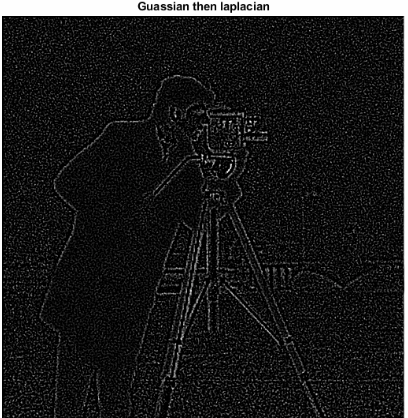
end

imshow(out);

title('Guassian then laplacian')



1. **CONCLUSIONS:**

Hence, I have successfully implemented our defined aims and objective, i.e. I have perform the unsharp masking by convolving the input image f(x, y)the average filter and then applied Laplacian operator on it and then also applied the Gaussian filter and vice vera.

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **Total Marks** | **Marks Obtained** | **Comments** |
| Designing Concept (D) | 3 |  |  |
| Application of Knowledge (E) | 2 |  |  |
| Performance(F) | 3 |  |  |
| Result (G) | 2 |  |  |
| Total | 10 |  |  |